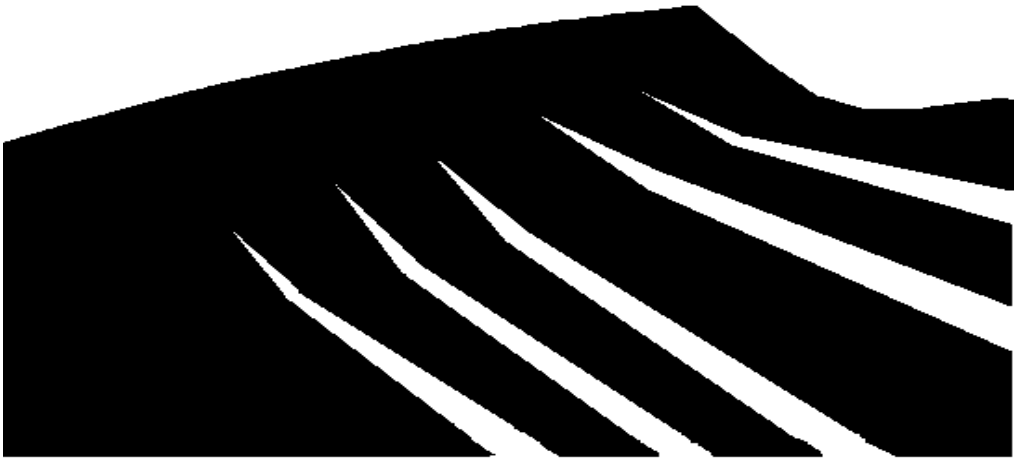
EFFECTIVE DATE August 8, 1996	LANL-YMP-QP-03.20, R5 Page 1 of 15

SOFTWARE CONFIGURATION MANAGEMENT

LOS ALAMOS

QUALITY PROGRAM



APPROVAL FOR RELEASE

B. D. GUNDLACH - PREPARER Signature on file	DATE Date on file
M. J. CLEVENGER - QUALITY ASSURANCE PROJECT LEADER Signature on file	DATE Date on file
J. A. CANEPA - TECHNICAL PROJECT OFFICER Signature on file	DATE Date on file

Los Alamos
Yucca Mountain Site Characterization Project

HISTORY OF REVISIONS

REVISION NO.	EFFECTIVE DATE	PAGES REVISED	REASON FOR CHANGE
R0	01/25/91	N/A	Initial procedure
R1	01/31/94	All	Complete rewrite to simplify process and incorporate QARD requirements.
R2	08/01/94	4, 5, 11, & 19	To incorporate RTN review comments and to specify personnel training.
R3	09/28/95	All	Complete rewrite to represent the flow of the developer and configuration manager in a simplified in-line approach.
R4	12/21/95	3, 4, & 15	Changes made to remain concurrent with the revised QARD.
R5	08/08/96	All	Minor non-substantive editorial changes and format changes to enhance readability.

SOFTWARE CONFIGURATION MANAGEMENT

1.0 PURPOSE

This procedure describes the organization, policies, procedures, and review process necessary to ensure the orderly control of the software products produced or acquired by the Los Alamos National Laboratory (Los Alamos) Yucca Mountain Site Characterization Project (YMP or Project).

2.0 SCOPE

2.1 This procedure governs software developed or acquired for use in support of licensing activities. It applies only to specifying, managing, evaluating, and certifying baselines that are the products of the software life-cycle specified in QP-03.21.

2.2 This procedure applies to all Los Alamos and Los Alamos-subcontractor YMP personnel (hereafter referred to as YMP personnel) who work under the Los Alamos YMP quality assurance program.

3.0 REFERENCES

DOE/RW-0333P, Office of Civilian Radioactive Waste Management Quality Assurance Requirements and Description (QARD)
LANL-YMP-QP-03.21, Software Life Cycle
LANL-YMP-QP-17.6, Records Management

4.0 DEFINITIONS

4.1 Reviews

Reviews are performed to assess the technical content of each proposed baseline and to ensure compliance at each stage of the life cycle. Reviews are performed under the auspices of the Configuration Control Board. Reviews provide an independent environment for the assessment of each product, identification of discrepancies or other review issues, and specification of the means and timetable by which review issues are resolved as follows:

- The Software Configuration Manager identifies the appropriate reviewers and distributes the review materials.
- The review leader manages the review and coordinates issue resolution with the developer.
- After of the review issues are resolved and approved by the Configuration Control Board, the Software Configuration Manager certifies the baseline.

4.2 Software Configuration Manager Controlled Datasets

The Software Configuration Manager controls datasets (computer storage space) where electronic copies of baseline submissions, software packages, and documentation are archived.

5.0 RESPONSIBILITIES

The following personnel are responsible for activities identified in Section 6.0 of this procedure:

- Software Configuration Manager (SCM)
- Configuration Control Board members (CCB)
- Review Leader
- Reviewers
- Developers

6.0 PROCEDURE

The use of this procedure must be controlled as follows:

- If this procedure cannot be implemented as written, YMP personnel should notify appropriate supervision. If it is determined that a portion of the work cannot be accomplished as described in this QP, or would result in an undesirable situation, that portion of the work will be stopped and not resumed until this procedure is modified or replaced by a new document that reflects the current work practice.

- YMP personnel may use copies of this procedure printed from the controlled document electronic file; however, YMP personnel are responsible for assuring that the correct revision of this procedure is used.

- When this procedure becomes obsolete or superseded, it must be destroyed or marked “superseded” to ensure that this document is not used to perform work.

NOTE: The software process is described in two procedures (QP-03.20 and QP-03.21). Personnel who want to propose changes to or new development of software normally initiate a Software Change (SC) by following QP-03.21. Personnel who need to use controlled software follow subsection 6.7 in QP-03.21.

6.1 Configuration Identification

6.1.1 The modules retain the module name specified by the developer. **SCM** generates a unique controlled dataset directory identifier using the identifying number of the associated Software Transmittal (ST), Attachment 3 of QP-03.21, directly relating the ST to the software items. The entire baseline transmittal is placed in the directory. This includes all documents, software modules, and file lists.

6.1.2 **SCM** records the release label of the sanctioned release in the comments field of the Certification Notice (CN), Attachment 1. For additional information regarding the CN, refer to subsection 6.2.3. Construct the release label as follows:

6.1.2.1 Obtain the application field from the name specified in the application's file list (AFL).

6.1.2.2 If this is the first release of the application, set the version number (vv) to 01, the revision number (rr) each to 00 (e.g., 01.00), and terminate the process.

6.1.2.3 If the release incorporates only minor functional changes, construct the remainder of the release label as follows:

- Retain the version number found in the release label of the most recent prior release of the product and assign this value to the version number field.

- Increment the revision number found in the release label of the most recent prior release of the product and assign this value to the revision number field.

6.1.2.4 If the release incorporates major functional changes (new or substantially revised interfaces, models, or logic for example), construct the remainder of the release label as follows:

- Increment the version number found in the release label of the most recent prior release of the product and assign this value to the version number field.

- Reset the revision number to 00 and assign this value to the revision number field.

6.1.2.5 Update the SCM controlled datasets by recording the release label as an attribute of the SCM controlled datasets identifier containing the corresponding release of the application.

6.2 Configuration Control

6.2.1 This subsection describes the process for software dissemination control and performing a release of information from the SCM controlled datasets. These datasets contain the archived baselines for Los Alamos SCM controlled certified and development software.

It is the responsibility of the SCM to ensure that software verification and validation of developed or modified software is performed prior to release. In those cases where this is not done prior to the release of software, written justification for the reason is normally documented in a memo,

where the portions of software that have not been verified and validated will be identified and controlled.

6.2.2 **SCM** processes a request for release of SCM controlled datasets information as follows:

6.2.2.1 Upon receipt of a ST:

- Verify that the form is signed by the originator.
- Verify that the software or document is available for release.
- Determine whether the information provided on the form is complete and consistent with the type of release being requested.
- Determine whether SCM has the authority to issue the release or whether CCB approval must be obtained.
- The **CCB** approves releases intended for distribution outside the Los Alamos YMP.

If verification fails, check the Not Accepted box in the Disposition field of the ST, and proceed to subsection 6.2.2.5.

6.2.2.2 Perform one of the following disposition actions based upon the outcome of subsection 6.2.2.1.

- If the information is complete and appropriate and the release can be performed as requested, check the Accepted box and proceed to subsection 6.2.2.5.
- If the release requires minor modifications to the request, contact the originator and update the ST form. Then if the release can be performed as requested, check the Accepted box, and proceed to subsection 6.2.2.4.

6.2.2.3 If CCB action is required:

a. Check the CCB Action box

b. Document the reason that CCB Action is required in the SCM / CCB Comments field

c. Forward the ST to the **CCB Chairperson** who will:

- Poll the CCB members or hold a meeting (if deemed necessary)
- Inform SCM of the CCB's decision

- Forward the ST back to SCM

- If the request is not approved for any other reason, check the Not Accepted box, document the reason in the SCM/CCB Comments field, and proceed to Step (d) of subsection 6.2.2.5.

6.2.2.4 Transfer the distribution package to the originator in accordance with the format and distribution specifications provided in the File Transfer Block of the ST.

6.2.2.5 Sign and date the ST, and notify the originator. Process the ST as a record, and terminate the process.

6.2.2.6 If problems arise during the installation and/or checkout of the software by the originator or prospective user, assist in identifying any features of the local installation environment that may be causing the problems.

6.2.3 **SCM** processes the Baseline Certification as follows:

6.2.3.1 After the review process for a baseline is completed, transfer the baseline into the archive SCM controlled datasets all components identified with the baseline on the task's Life Cycle (LC) form, Attachment 7 of QP-03.21.

6.2.3.2 Document the baseline certification process on a CN as follows:

- General Information. Record the certification date in the Date field and complete the title field. Use the contents of the title field of the corresponding LC. Provide the name of the Developer, the Developer's organization, telephone, and mail stop in the fields provided.

- LC Reference. Provide the identifier of the LC that controlled the development of the baseline.

- Baseline Identification. Check the appropriate box(es) to identify the baseline that is being certified. If certification follows the completion of a baseline closure review, check the corresponding Baseline/Phase Closed box.

- Furnish any additional comments in the Comments field.

- Print the SCM representative’s name, sign and date the form in the appropriate fields.

6.2.3.3 Upon completion of the certification process the developer is notified. One copy of the CN should be placed in the CN folder. The CN is also processed as a record.

6.2.3.4 Upon sanctioning a new version of a previously sanctioned software application, notify the registered users of the application that it is available for dissemination. Registered users of a sanctioned application are those individuals for whom an approved ST (or Software/Data Dissemination Request to provide continuity with previous Los Alamos Yucca Mountain Project Software Quality Assurance programs) is on file with the SCM.

NOTE: Registered users notification may be done by voice, electronic or hard copy mailing.

6.3 Reviews

Effective reviews provide the way to ensure quality and document that all developed or modified Los Alamos YMP software is validated and verified prior to release.

6.3.1 All reviews of software baselines are authorized and approved by the **CCB**. If at some point subsequent to review approval, the **CCB** or the **development organization** determines a need for additional reviews. The **CCB Chairperson** authorizes these reviews.

6.3.2 Upon receipt of the baseline submission ST form from the Developer, the **SCM** transfers the proposed baseline to the SCM controlled datasets. The **SCM** also files the ST into a folder into which a copy of all hard copy documentation associated with the baseline will be collected for later submission as a records package.

6.3.3 The **CCB Chairperson** identifies the Review Leader who will head the review committee. The **Review Leader** is the person or persons who verify that all review issues have in fact been resolved or addressed. This person must not have worked on the original development or modification of the software under review.

The **Review Leader** manages the review as follows:

- Maintains emphasis on identification of quality affecting problems, rather than discussions of potential solutions.
- Ensures that the review progresses through the baseline documentation in an orderly fashion.
- Ensures that all quality affecting items and issues identified by the reviewers are documented on Software Review (SR) form, Attachment 2.
- Based on the scope and quantity of the comments, the Review Leader may call the reviewers together for a review meeting.

6.3.4 The **SCM** identifies reviewers that will participate in the review through coordination with the Review Leader.

6.3.4.1 For verification and validation (V&V) related reviews, reviewers must not have worked on the original code development or modification. These V&V related reviews may be performed by the person who directed the code development or modification task with approved and documented justification from higher level group management. A person performing V&V documentation development and test activities may also be a V&V related reviewer if these activities are independent of the code development or modification task.

6.3.4.2 Permanent CCB members are automatically eligible to participate in all reviews, but they are not necessarily expected to be technical experts. If additional technical expertise is needed, the **CCB Chairperson** may identify temporary CCB members to participate in the review.

6.3.4.3 **Reviewers** coordinate all their review activities with the Review Leader.

6.3.5 The **SCM** coordinates the review schedule with the Review Leader, prepares one review packet for each review participant and distributes the packets. The review packet becomes part of the record package associated with the baseline. Review packets consist of a soft (or information where the review dataset can be accessed) or hard copy of the entire developer-submitted baseline and a copy of the SR form prepared as follows:

- Application. Fill in the name of the application. (e.g., GZSOLVE").
- Review: Indicate the baseline being reviewed and the review type (e.g., "Detailed Design Baseline In-process review").
- Title: Enter the full name of the application. (e.g., "GZSOLVE Reuse Component")
- Developer Information: Fill in the developer's name and phone in the appropriate areas.
- SR number: SR form identifiers include the (SC), Attachment 1 of QP-03.21, LC, and associated ST number of the submission package for the development or maintenance task (e.g., **SR**-*10-2-5*-21: This is the twenty-first software review form associated with software transmittal five of life cycle 2 of software change 10.).
- SCM Representative: The **SCM** signs on this line.
- Date: Issue date:

- Review Leader: Fill in the review leader's name and phone number.

- Comments Due Date: The **SCM** coordinates this date with the Review Leader.

- **Reviewer**: Enter the name and phone number of a reviewer in the Reviewer name and phone area of the Reviewer Comments section.

6.3.6 **Reviewers** must address the software verification and validation issues associated with the various life cycle phases as described below.

6.3.6.1 As part of the review process, evaluate whether traceability requirements are met by the proposed baseline.

a. Verify traceability of the software requirements into the software design as follows. If no Software Requirements Specification is specified for the development task, proceed to Step (b) of this subsection.

Ensure that each requirement in the Software Requirements Specification is implemented within one or more modules of the software design (as embodied in the documentation prolog of each software module).

b. Establish that the requirements specified in the Software Requirements Specification are traceable into the software implementation (code). Verify that the source code is a faithful rendition of the detailed design by comparing the source code to the corresponding pseudocode.

c. If Steps (a) or (b) of this subsection expose significant requirements traceability errors, note the errors on the SR form.

6.3.6.2 Software V&V Plans describe methods (such as review, inspection, analysis, demonstration and test) for verification and validation.

- Reviews are the primary method of software baseline verification.

- Testing and demonstration are the primary method of software validation.

- Software validation of modifications to released software items includes regression testing.

As part of the review process, evaluate whether the testing program for the proposed baseline is satisfactory. Note any problems for discussion during the review.

6.3.7 The **Reviewer** documents review issues in the Reviewer Comments section of the SR form. Each comment includes:

- Module name and affected document
- Issue - describe the issue emphasizing identification of quality affecting problems, rather than discussions of potential solutions. (Section I of Attachment 4 demonstrates an example of a document and code issue comment)

NOTE: Any editorial suggestions (suggested grammatical changes, style issues, etc.) should be kept at a minimum.

- The **reviewer** certifies his or her comments by signing and dating the signature and date areas of the Reviewer Comment section.
- Return all review comments to the review leader.

6.3.8 If review issues are identified, the **Review Leader** delivers the completed SR forms to the Developer for issue resolution. Resolve all issues to the satisfaction of the reviewer. The **Developer** documents his or her response in the Resolution section of the SR form. (Section II of Attachment 4 demonstrates an example of a document and code issue response)

6.3.9 After the **Review Leader** and **Reviewer** all agree that all issues have been satisfactorily resolved by the Developer, they certify their agreement by signing and dating their corresponding areas in the Closure section of the SR form.

6.3.10 After all review issues have been closed, the **Review Leader** forwards the completed SR forms to the CCB Chairperson. The **CCB Chairperson** then polls the CCB membership to approve and finalize the review. Each CCB member gets one vote. The **CCB** approves the review by a majority vote.

If the review is approved, the **CCB Chairperson** documents the decision in meeting minutes in the form of a signed memo from the CCB Chairman to SCM. The **CCB Chairperson** forwards the completed SR forms and the minutes memo to the SCM for inclusion in the baseline record package.

NOTE: There is no requirement for a formal meeting to vote. The CCB Chairperson may poll the review participants by e-mail, telephone, mail, in-person. Most importantly, all quality affecting review issues are documented as resolved.

If the review is not approved, the baseline (with a memo from the CCB Chairperson detailing the reason for disapproval) is returned to the Developer for modification. The **Developer** (upon completion of modifications) submits a revised baseline to SCM. The **SCM** schedules a subsequent review when the baseline is resubmitted.

All CCB decisions are documented in meeting minutes in the form of a signed memo from the **CCB Chairman** to SCM.

6.4 Configuration Status Accounting

6.4.1 Assign and record a unique identifier in the identification field of the form.

The identifier includes the form type and serial number. **SCM** assigns numbers to all forms except the Continuation Form -- a **developer, analyst, CCB member, SCM, Review Leader, reviewer, or originator** may assign the number provided the criteria below are followed.

1 The form you are continuing identification type (SC, SR, LC, ST, or CN)

2 The form you are continuing identifying number which follows the identification type. The number was assigned by SCM.

3 Followed by a continuation form sequence number. (Use an ascending sequence beginning with "1" for all CF forms associated with the form you are continuing.)

(e.g., CF-*SC-10*-1. The italicized part of this example comes directly from the form you are continuing. This identifies the first continuation form for software change 10.)

(e.g., CF-*SR-10-5-2-21*-2. The italicized part of this example comes directly from the form you are continuing. This identifies the second continuation form for the twenty-first software review form associated with software transmittal five of life cycle 2 of software change 10.)

- SC forms are the primary identifier for a development or maintenance task. (e.g., SC-10. This is software change 10.)

- LC form identifiers include the SC number of the development or maintenance task. (e.g., LC-*10*-2. This is the second life cycle form associated with software change 10.)

- ST form identifiers include the SC and LC number of the development or maintenance task. (e.g., ST-*10-2*-5. This is the fifth software transmittal form associated with life cycle 2 of software change 10).

- SR forms form identifiers include the SC, LC, and associated ST number of the submission package for the development or maintenance task.

- (e.g., SR-*10-2-5*-21. This is the twenty-first software review form associated with software transmittal five of life cycle 2 of software change 10.)

- CN forms form identifiers include the SC and LC number of the development or maintenance task.

- (e.g., CN-*10-2*-6. This is the sixth certification form associated with life cycle 2 of software change 10.)

6.4.2 **SCM** provides the capability to identify the current configuration status of any certified software baseline and to track changes by establishing the following minimum suite of Configuration Status Accounting reports:

- A summary listing of the current approved configuration showing all certified modules and their interrelationships.

- A summary listing of the status of proposed changes to the approved configuration.

- A brief chronology of versions of each sanctioned software application including a summary of the changes that produced each new version using the Version Description Documents associated with the software.

- A cross-reference between baseline documents and associated software in the form of the files list associated with the baseline.

6.5 Media Control

SCM controls the SCM controlled datasets as follows:

- Access Control. Employ password controls to restrict access to the SCM controlled datasets to the SCM.

- Archive soft copies of baselines on SCM controlled datasets.

7.0 RECORDS

7.1 Project QA Record Packages

This procedure implements three distinct record packages that compose the complete suite of QA documentation for software baselines and the Software Quality Assurance effort. The **SCM** produces and submits each record package to

a Los Alamos YMP Records Processing Center in accordance with the provisions of QP-17.6.

7.1.1 The certified baseline record-package documents all aspects of the development, assessment, and certification of a baseline. It is produced by the SCM after it formally certifies (issues a Baseline Certification Notice for) the subject baseline. It comprises the following components:

- The baseline components specified in subsection 6.5.2 of QP-03.21 for the corresponding baseline.
- Any ST forms that were issued for software releases or baseline submissions for review.
- All SR forms that describe the formal audits and reviews to which the baseline was subjected.
- The CN form that documents the certification of the baseline.
- All Continuation Forms (if any) referenced by any of the above supplemental documentation forms.
- The CCB Meeting Minutes documenting CCB decisions.
- A copy of the associated SC form if available.
- A copy of the associated LC form if available.

7.1.2 The software dissemination record-package describes and documents the disposition of a formal request to release sanctioned Los Alamos YMP software to an external entity. It is produced by the SCM after the subject ST is closed. It consists of the following closed supplemental documentation forms:

- One or more STs.
- All Continuation Forms referenced by any of the above supplemental documentation forms.

7.1.3 The SCM record package describes and documents the routine operation of the SCM. It is produced as required by the **Software Configuration Manager.**

It consists of any CSA Reports generated and distributed since the last record package was produced and any memos written to specify SCM activities and policy decisions that are not documented in others forms or reports.

8.0 TRAINING

8.1 Prior to conducting work described in Section 6.0, the Software Configuration Manager, members of the CCB, developers, review leaders, and reviewers require training to this procedure.

8.2 Training to this procedure is accomplished by "read only."

9.0 ATTACHMENTS

Attachment 1: Certification Notice (1 page)
Attachment 2: Software Review (1 page)
Attachment 3: Continuation Form (1 page)
Attachment 4: Documentation of Review Issues (1 page)

CERTIFICATION NOTICE

DATE: _____ **CN -** _____

TITLE: _____

DEVELOPER: _____

ORGANIZATION: _____ PHONE: _____ MAIL STOP: _____

REQUIREMENTS PHASE:

☐ REQUIREMENTS BASELINE ☐ BASELINE/PHASE CLOSED

DESIGN PHASE:

☐ PRELIMINARY DESIGN BASELINE

☐ V&V PLAN BASELINE

☐ DETAILED DESIGN ☐ BASELINE/PHASE CLOSED

IMPLEMENTATION PHASE:

☐ V&V PROCEDURE BASELINE

☐ IMPLEMENTATION BASELINE ☐ BASELINE/PHASE CLOSED

COMMENT:

SCM
REPRESENTATIVE: _____ _____ _____
Print name Signature Date

SOFTWARE REVIEW

APPLICATION: _____ REVIEW: _____

TITLE: _____

DEVELOPER'S NAME: _____ PHONE: _____ SR- _____

SCM REPRESENTATIVE: _____ DATE: _____

REVIEW LEADER: _____ PHONE: _____ COMMENTS DUE BY: _____

REVIEWER COMMENTS:

REVIEWER: _____ (Print name) _____ (Signature) _____ (Phone) _____ (Date)

RESOLUTION:

CLOSURE: REVIEWER: _____ (Signature) _____ (Date)

REVIEW LEADER: _____ (Signature) _____ (Date)

CONTINUATION FORM

ORIGINATOR: _____ DATE: _____

ORGANIZATION: _____ PHONE: _____ MAIL STOP: _____

CF- _____

COMMENTS

Los Alamos
Yucca Mountain Site
Characterization Project

Documentation Of Review Issues

Section I. Document and Code Issue Comment

EXAMPLE: Module: VVP

1) Expand on discussion of multiple degrees of freedom in last paragraph of Section 2.0. Inclusion of an example or figure / diagram of a multiple degree of freedom matrix might simplify / enhance understanding of the concept.

2) Include the name of the test program in Section 3.0.

Module: GZSOLVE TEST

1) Add command at start of script which prints descriptive header and date at start of test results file, i.e., "echo `GZSOLVE APPLICATION TEST RESULTS' >> gzsolve.tr; date >> gzsolve.tr"

2) Comment on commands in test script that are specific to platform script is being executed on.

Section II. Document and Code Issue Response

EXAMPLE: Module: VVP

1) DONE in 4.1.4.1.4 , then referred to later. - I did this the first time they are mentioned, then referred back to this section. Did not put in Section 3. (didn't belong there)

Module: GZSOLVE TEST

1) DONE
2) DONE